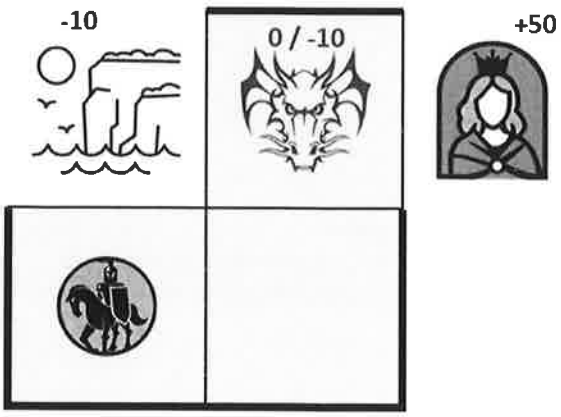
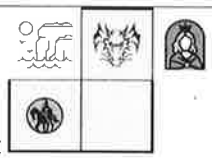
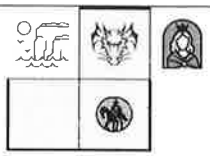
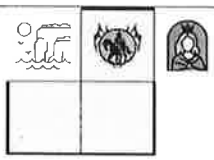
Apprentissage d'un labyrinthe avec le Q-learning

Un chevalier se trouve dans un labyrinthe de 3 cases. Il se déplace en choisissant une des quatre directions cardinales. Vous allez exécuter à la main toutes les étapes de calcul de l'algorithme Q-learning au cours d'un apprentissage du déplacement dans ce labyrinthe !

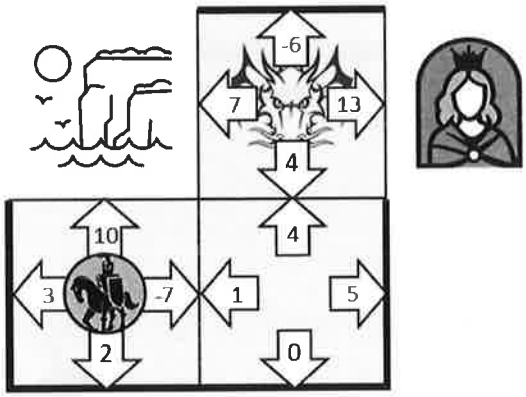


Voici comment fonctionnent les déplacements et les récompenses :
- si le chevalier essaie de franchir un mur épais, il reste dans la même case et n'obtient aucune récompense.
- s'il se déplace vers la princesse, il est récompensé de +50 points et une nouvelle partie recommence (il est placé sur la case en bas à gauche).
- s'il se déplace vers la falaise, il meurt, perd -10 points, et une nouvelle partie recommence.
- s'il se déplace vers la case du dragon un combat s'engage : avec une chance sur deux le chevalier reste vivant, il n'obtient aucune récompense et reste sur la case ; avec une chance sur deux il meurt, perd -10 points, et la partie recommence.

La table du Q-learning récapitule les « valeurs » assignées à chacune des 4 actions à partir de chacun des 3 états possibles. On l'initialise aléatoirement (avec des valeurs qui ne sont donc pas correctes).

« Q-valeur » de l'action a depuis l'état s	Action ↑	Action →	Action ↓	Action ←
Etat 	10	-7	2	3
Etat 	4	5	0	1
Etat 	-6	13	4	7

Nous allons représenter cette table sous la forme plus agréable suivante :



La partie commence : l'IA choisit l'action de plus grande valeur d'après sa table : 

. Le chevalier tombe de la falaise et reçoit la récompense R = -10, la partie s'arrête. Nous utilisons la formule du Q-learning pour mettre à jour la valeur $Q(s_t, a_t)$ de l'action qui vient d'être effectuée :

$$target = R_t + \gamma \max_a Q(s_{t+1}, a)$$

$$Q(s_t, a_t) \leftarrow (1 - \alpha) Q(s_t, a_t) + \alpha\, target$$

Pour simplifier nous choisissons comme *vitesse d'apprentissage* $\alpha = 0.5$ et *facteur d'actualisation* $\gamma = 0.5$, et *lors des calculs nous ne garderons pas les nombres après la virgule, mais arrondirons à l'entier le plus proche* (si on a .5 après la virgule, on arrondit dans la direction de zéro, par exemple -3.5 → -3).

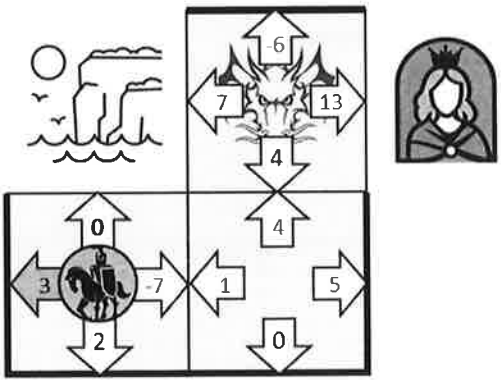
Comme la partie s'arrête avec la mort du chevalier, il n'y a pas « d'état suivant » s_{t+1}, donc pour cette fois-ci la partie $\gamma \max_a Q(s_{t+1}, a)$ de l'équation est ignorée :

$$target = R_t = -10$$

$$Q(s_t, a_t) \leftarrow (1 - \alpha) * Q(s_t, a_t) + \alpha * target = 0.5 * 10 + 0.5 * (-10) = 0$$

La Q-valeur est donc remplacée par 0. Une nouvelle partie recommence, l'IA choisit à nouveau l'action de plus grande valeur, mais à présent c'est la direction ouest 

.



Le chevalier reçoit une récompense de 0 et reste sur la même case. Nous calculons la mise à jour de la Q-valeur :

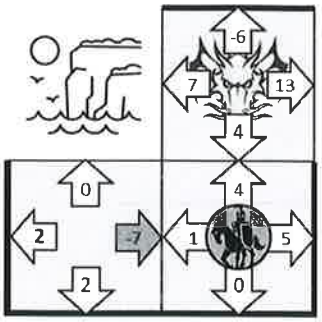
$$target = R_t + \gamma \max_a Q(s_{t+1}, a) = 0 + 0.5 * 3 = 1.5$$

$$Q(s_t, a_t) \leftarrow (1 - \alpha) Q(s_t, a_t) + \alpha\, target = 0.5 * 3 + 0.5 * 1.5 = 1.5 + 0.75 = 2.25$$

On arrondit à 2.

A présent l'IA fait faire une *exploration* au chevalier en choisissant l'action (non préférée) 

.



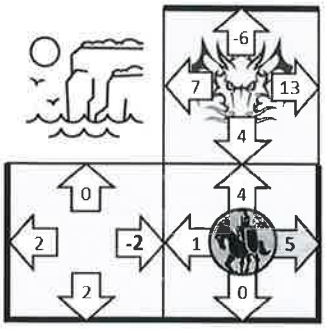
Récompense reçue : 0.

Calcul de la mise à jour :

$target = R_t + \gamma \max_a Q(s_{t+1}, a) = 0 + 0.5 * 5 = 2.5$

$Q(s_t, a_t) \leftarrow 0.5 * (-7) + 0.5 * 2.5 = -3.5 + 1.25 = -2.25$

On arrondit à -2.



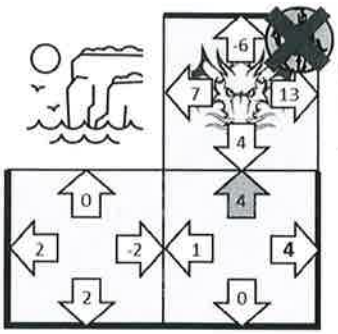
L'IA choisit ensuite l'action de plus forte valeur 

Récompense reçue : 0

$target = R_t + \gamma \max_a Q(s_{t+1}, a) = 0 + 0.5 * 5 = 2.5$

$Q(s_t, a_t) \leftarrow 0.5 * 5 + 0.5 * 2.5 = 2.5 + 1.25 = 3.75$

On arrondit à 4.



Action choisie : 

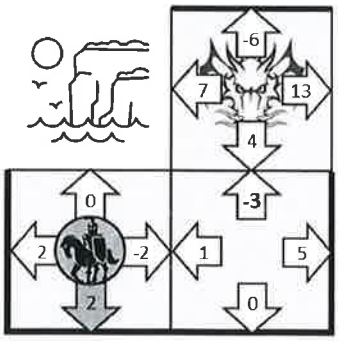
Le chevalier arrive sur la case du dragon, l'issue du combat lui est fatale, il meurt et perd -10 points. Fin de partie.

$target = R_t = -10$

$Q(s_t, a_t) \leftarrow 0.5 * 4 + 0.5 * (-10) = 2 - 5 = -3$

Le chevalier repart de la case sud-ouest.

A vous de faire les calculs maintenant !



Action choisie : 

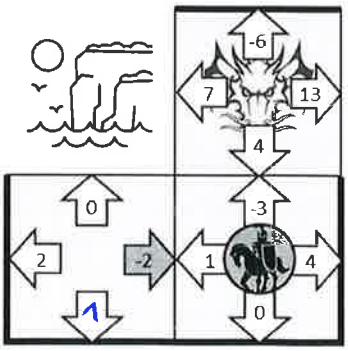
Le chevalier reste sur la même case, récompense : 0

$target = R_t + \gamma \max_a Q(s_{t+1}, a) =$ $0 + \frac{1}{2} \cdot 2 = 1$

$Q(s_t, a_t) \leftarrow 0.5 * Q(s_t, a_t) + 0.5 * target =$ $\frac{1}{2} \cdot 2 + \frac{1}{2} \cdot 1 = 1,5$

On arrondit à 1

Désormais, ajoutez les valeurs manquantes dans les flèches suite aux apprentissages



Action choisie : 

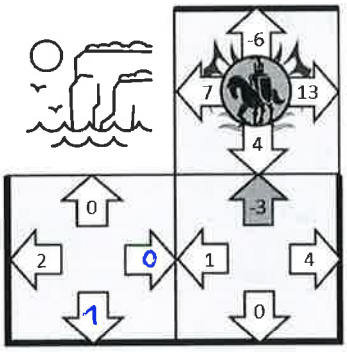
(exploration)

Récompense : 0

$target = 0 + \frac{1}{2} \cdot 4 = 2$

$Q(s_t, a_t) \leftarrow \frac{1}{2}(-2 + 2) = 0$

On arrondit à 0



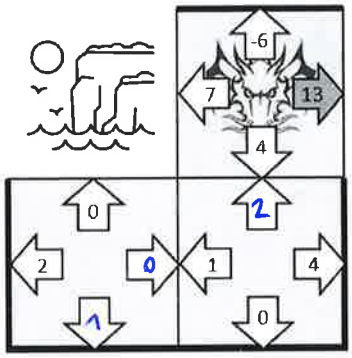
Action choisie :  (exploration ; oui il y a beaucoup d'explorations heureuses, sinon ce serait trop long !)

Nouveau combat contre le dragon, cette fois la chance sourit au chevalier, il reste en vie et reste sur la case. Récompense : 0.

$target = 0 + \frac{1}{2}\,13 = 6.5$

$Q(s_t, a_t) \leftarrow \frac{1}{2}(-3 + 6.5) = 1.75$

On arrondit à 2



Action choisie : 

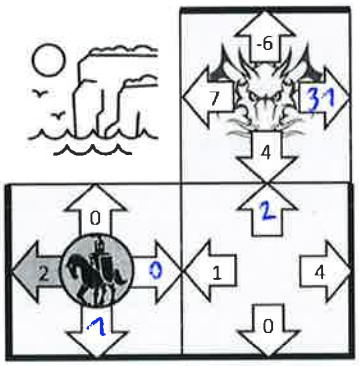
Le chevalier a rejoint la princesse ! Fin de partie. Récompense : +50.
Comme d'habitude, on met à jour la valeur de l'action :

$target = R_t = +50$

$Q(s_t, a_t) \leftarrow \frac{1}{2}(13 + 50) = 31.5$

On arrondit à 31

Une nouvelle partie commence sur la case sud-ouest.



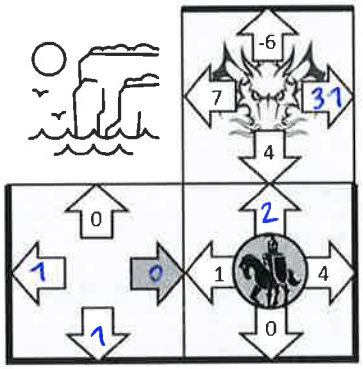
Action choisie : 

Le chevalier reste sur la même case. Récompense : 0.

$target = 0 + \frac{1}{2}\,2 = 1$

$Q(s_t, a_t) \leftarrow \frac{1}{2}(2 + 1) = 1.5$

On arrondit à 1



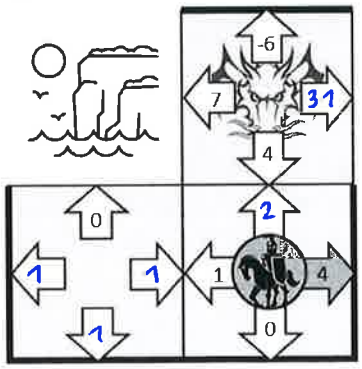
Action choisie :  (encore une exploration heureuse)

Récompense : 0

$target = 0 + \frac{1}{2} \cdot 4 = 2$

$Q(s_t, a_t) \leftarrow \frac{1}{2}(0 + 2) = 1$

On arrondit à 1



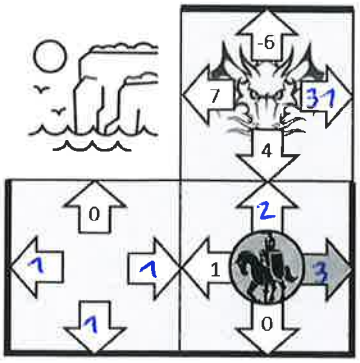
Action choisie : 

Récompense : 0. On reste sur la même case.

$target = 0 + \frac{1}{2}\,4 = 2$

$Q(s_t, a_t) \leftarrow \frac{1}{2}(4 + 2) = 3$

On arrondit à 3



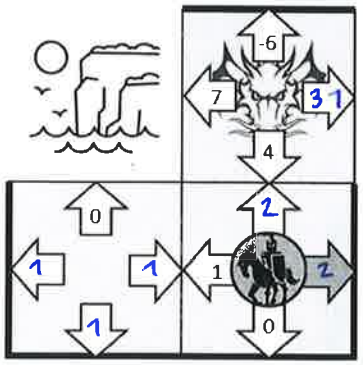
Même action choisie : 

Récompense : 0. On reste sur la même case.

$target = 0 + \frac{1}{2}3 = 1.5$

$Q(s_t, a_t) \leftarrow \frac{1}{2}(3 + 1.5) = 2.25$

On arrondit à 2



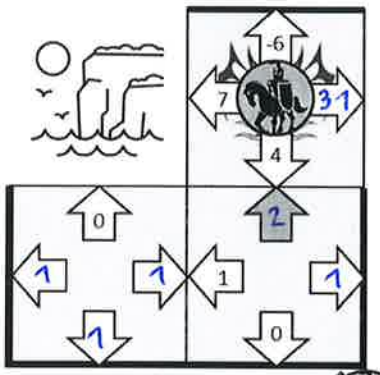
Même action choisie : 

Récompense : 0. On reste sur la même case.

$target = 0 + \frac{1}{2} \cdot 2 = 1$

$Q(s_t, a_t) \leftarrow \frac{1}{2}(2 + 1) = 1.5$

On arrondit à 1



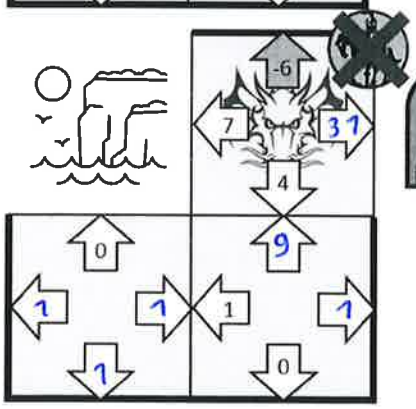
Action choisie : 

Nouveau combat contre le dragon. Le chevalier résiste à nouveau.
Récompense : 0.

$target = 0 + \frac{1}{2} \cdot 31 = 15.5$

$Q(s_t, a_t) \leftarrow \frac{1}{2}(2 + 15.5) = 8.75$

On arrondit à 9



Action choisie :  (exploration... non favorable cette fois)

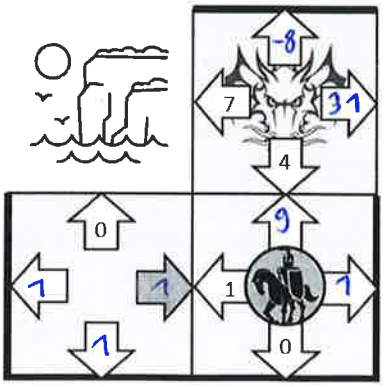
Le chevalier reste sur la même case, doit combattre à nouveau le dragon, et cette fois meurt. Récompense : -10. Fin de partie.

$target = R_t = -10$

$Q(s_t, a_t) \leftarrow \frac{1}{2}(-6 - 10) = -8$

On arrondit à -8

Le chevalier repart de la case sud-ouest.



Action choisie : 

Récompense : 0.

$target = 0 + \frac{1}{2} \cdot 9 = 4.5$

$Q(s_t, a_t) \leftarrow \frac{1}{2}(1 + 4.5) = 2.75$

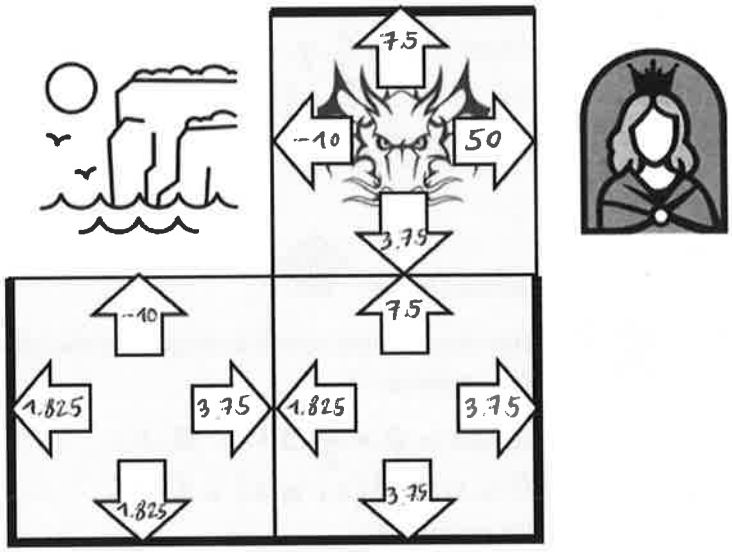
On arrondit à 3

Désormais, les Q-valeurs estimées reflètent bien mieux la vraie valeur de chaque action, et en particulier si le chevalier ne fait plus d'exploration, il ira toujours à la princesse en partant de la case sud-ouest (pourvu qu'il ne soit pas arrêté par le dragon).

Question facultative (plus difficile) : convergence de l'apprentissage du labyrinthe

Si on continue l'apprentissage, les Q-valeurs vont osciller, notamment celles des actions qui arrivent sur la case dragon, car le calcul de $target$ dépend de l'issue aléatoire du combat entre le chevalier et le dragon.

Mais si on diminue la vitesse d'apprentissage α, ces oscillations pourront être petites, et les valeurs de la table Q finiront par converger au bout d'un grand nombre d'actions et de mises à jour. Seriez-vous capable de déterminer les valeurs au terme de cette convergence ? Notez les ci-dessous (on garde $\gamma = 0.5$; cette fois, déterminez les valeurs exactes, sans arrondir à l'entier le plus proche):



Note : au terme de la convergence, les Q-valeurs vérifient l'équation de Bellman :

$$Q(s_t, a_t) = E(R_t + \gamma \max_a Q(s_{t+1}, a)),$$

où E désigne l'espérance mathématique.

5. Analyse de document : ChatGPT

Le 30 novembre 2022, la fondation OpenAI a mis en ligne un chatbot conversationnel nommé « ChatGPT », capable de discussions époustouflantes. (Je vous encourage de retour chez vous à l'essayer : Cherchez sur Internet « ChatGPT » ; vous pourrez également essayer « Dall-e » qui génère des images automatiquement à partir de descriptions textuelles)

Vous trouverez ci-joint la page de blog https://openai.com/blog/chatgpt/ et sa traduction et légère simplification en français. Nous allons voir comment, suite au cours Découverte de l'IA, vous êtes capable de comprendre les quelques descriptions techniques sur comment a été entraîné ChatGPT ! Comme vous le verrez en effet ces explications exigent déjà un certain niveau de connaissance pour pouvoir être comprises, elles ne disent pas forcément tout, et sont certainement seulement une simplification du vrai processus ; les questions ci-dessous vont vous guider.

Cochez les bonnes réponses ci-dessous. Pour certaines questions il faut cocher plusieurs bonnes réponses.

L'article parle beaucoup de « modèles ». Un modèle en Machine Learning est comme une « fonction programmable ». Par exemple, un réseau de neurones est un modèle. Un modèle génère des sorties en fonction d'entrées qu'on lui donne. Un modèle a des paramètres qui peuvent être modifiés par un algorithme d'apprentissage. Dans le cours nous avons parlé de classifieur plutôt que de modèle : un classifieur est une catégorie particulière de modèle qui comme son nom l'indique, fait de la classification (ses sorties sont des numéros de catégorie).

Etant donné cette définition d'un modèle en Machine Learning, est-ce que :

- ☐ l'algorithme des K plus proches voisins est un modèle
- ☒ un réseau de neurone est un modèle
- ☒ la table du Q-learning est un modèle

ChatGPT est :

- ☒ une IA
- ☒ un agent conversationnel (chatbot)
- ☒ un modèle de Machine Learning

ChatGPT est capable de :

- ☒ répondre à des questions de culture générale
- ☒ inventer des poèmes
- ☒ se rappeler des précédentes questions qu'on lui a posées et de ses précédentes réponses

Qu'est-ce que le modèle ChatGPT prend en entrée ?

- ☒ la dernière phrase tapée par l'utilisateur, mais également l'historique des questions et réponses précédentes
- ☐ la dernière phrase tapée par l'utilisateur uniquement
- ☐ une très grande base de donnée de questions et de réponses

Qu'est-ce que le modèle ChatGPT renvoie comme sorties ?

- ☒ des réponses sous forme d'un texte d'un ou plusieurs paragraphes
- ☐ des nombres

Contrairement aux exemples d'apprentissages vus en cours, ChatGPT n'a pas été entraîné en utilisant un seul algorithme et une seule base de données, mais selon une procédure complexe en plusieurs étapes, et mettant en jeu plusieurs algorithmes et plusieurs bases de données. Il est une évolution du modèle GPT-3.5 qui a été entraîné auparavant. Le texte parle également d'un autre modèle appelé InstructGPT. Est-ce que :

- ☐ ChatGPT est une évolution de InstructGPT, qui est lui-même une évolution de GPT-3.5?
- ☒ ou ChatGPT et InstructGPT sont tous les deux des évolutions de GPT-3.5, mais aucun n'est une évolution de l'autre ?

Dans le cadre des réseaux de neurones vus en cours pour piloter le robot AlphAI, serait-il envisageable d'entraîner un tel réseau avec de l'apprentissage supervisé, puis de poursuivre cet entraînement avec de l'apprentissage par renforcement ?

- ☒ oui : les connexions dans le réseau seront modifiées tout d'abord par l'apprentissage supervisé, puis par l'algorithme d'apprentissage par renforcement (par exemple par DQN étudié en cours)
- ☐ non : au démarrage de l'apprentissage par renforcement les connexions sont forcément réinitialisées aléatoirement et on ne peut pas garder de trace d'un premier apprentissage supervisé

Le texte joint n'explique pas comment GPT-3.5 a été entraîné, mais on trouve ailleurs sur Internet que GPT-3.5 a été entraîné pour compléter automatiquement le début d'un texte. A votre avis, grâce à quel type de base de donnée GPT-3.5 a été entraîné ?

- ☐ à partir d'un grand nombre de textes d'exemples écrits par des entraîneurs humains spécialement pour entraîner GPT-3.5
- ☒ à partir d'un maximum de textes déjà disponibles (Wikipedia, encyclopédies, livres scannés, sites Internet, etc.)

L'étape 1 décrite dans le texte et dans la figure est une première étape d'amélioration de GPT-3.5, en quoi consiste-t-elle ?

- ☒ un apprentissage supervisé utilisant comme données des exemples de réponses fournis par un entraîneur humain
- ☐ un apprentissage supervisé utilisant comme données des réponses générées par GPT-3.5

Comment fonctionnent les étapes 2 et 3 ?

- ☐ dans l'étape 2, la première version de ChatGPT issue de l'étape 1 est améliorée avec de l'apprentissage par renforcement (c'est un humain qui donne des récompenses) ;
  dans l'étape 3, elle est encore améliorée, toujours avec de l'apprentissage par renforcement (mais cette fois les récompenses sont générées de manière automatique)
- ☒ dans l'étape 2, un nouveau modèle auxiliaire est entraîné à donner des notes aux réponses de ChatGPT ;
  dans l'étape 3, la première version de ChatGPT issue de l'étape 1 est améliorée avec de l'apprentissage par renforcement (les récompenses sont déterminées par le modèle auxiliaire de l'étape 2)

6. Apprentissage d'un labyrinthe avec le Q-learning

Un chevalier se trouve dans un labyrinthe de 3 cases. Il se déplace en choisissant une des quatre directions cardinales. Vous allez exécuter à la main toutes les étapes de calcul de l'algorithme Q-learning au cours d'un apprentissage du déplacement dans ce labyrinthe !